MICROCHIP

Section 29. Real-Time Clock and Calendar (RTCC)

HIGHLIGHTS

This section of the manual contains the following major topics:

> **Note:** This family reference manual section is meant to serve as a complement to device data sheets. Depending on the device variant, this manual section may not apply to all dsPIC33E/PIC24E devices.
>
> Please consult the note at the beginning of the **"Real-Time Clock and Calendar (RTCC)"** chapter in the current device data sheet to check whether this document supports the device you are using.
>
> Device data sheets and family reference manual sections are available for download from the Microchip Worldwide Web site at: http://www.microchip.com

29.1 INTRODUCTION

This section describes the Real-Time Clock and Calendar (RTCC) module that provides a full Binary-Coded Decimal (BCD) clock calendar. Some of the key features of the RTCC module are:

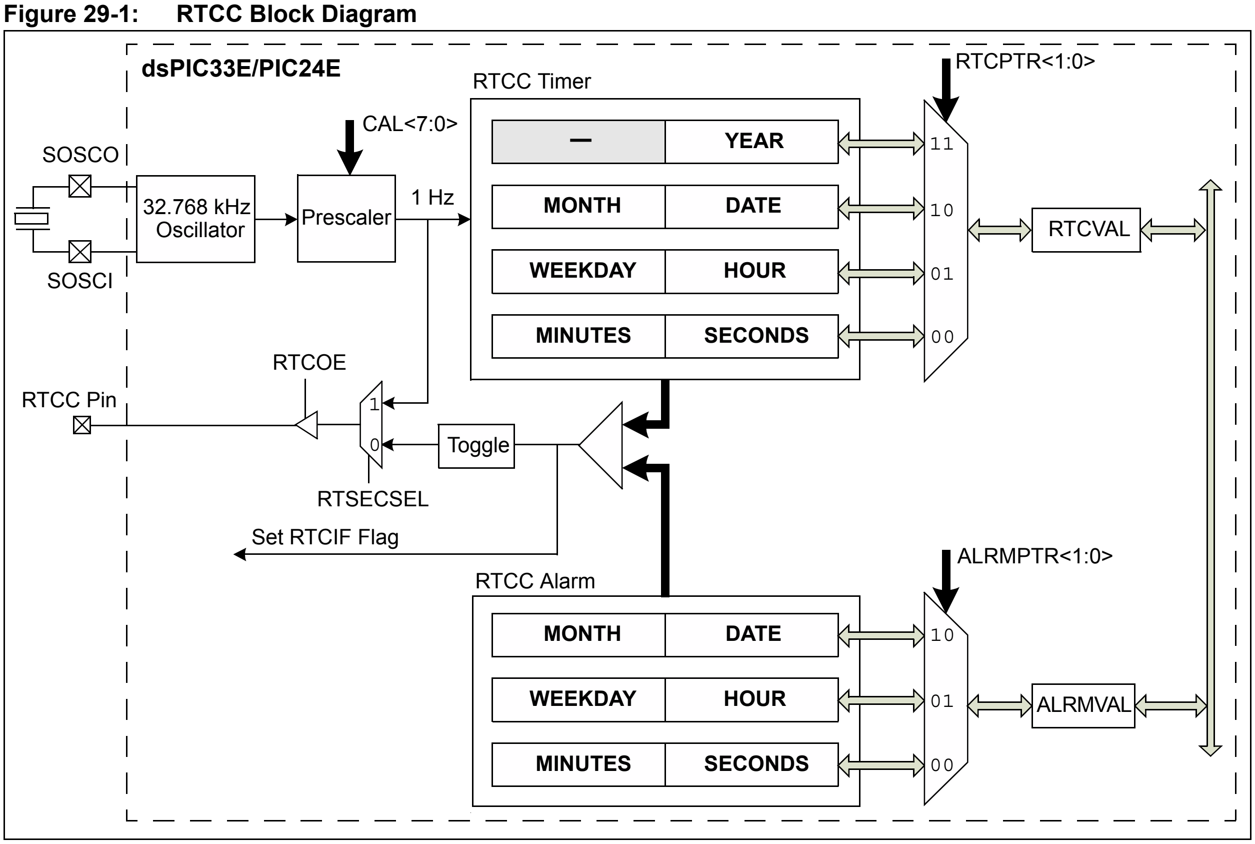
- 24-hour (military time) clock
- 100-year calendar up to year 2099
- Counts seconds, minutes, hours, weekday, date, month and year; with leap year compensation
- BCD representation of time, calendar and alarm
- Programmable alarm with Repeat mode
- Square wave generation using alarm or 1 Hz clock output on RTCC pin
- RTCC calibration

The RTCC module provides a time reference to an application running on the device with minimum to no intervention from the CPU. The current date and time is tracked in a set of counter registers that are updated once per second.

The RTCC and the Secondary Oscillator (SOSC) continue to function when the device is held under Reset by pulling the $\overline{\text{MCLR}}$ pin low.

Figure 29-1 illustrates the block diagram of the RTCC module.

Figure 29-1: RTCC Block Diagram



29.2 RTCC MODULE REGISTERS

The RTCC module registers are organized into three categories:

- RTCC Control Registers
  - RCFGCAL: RTCC Calibration and Configuration Register
  - PADCFG1: Pad Configuration Control Register
  - ALCFGRPT: Alarm Configuration Register
- RTCC Value Registers
  - RTCVAL (when RTCPTR<1:0> = `11`): Year Value Register
  - RTCVAL (when RTCPTR<1:0> = `10`): Month and Day Value Register
  - RTCVAL (when RTCPTR<1:0> = `01`): Weekday and Hours Value Register
  - RTCVAL (when RTCPTR<1:0> = `00`): Minutes and Seconds Value Register
- Alarm Value Registers
  - ALRMVAL (when ALRMPTR<1:0> = `10`): Alarm Month and Day Value Register
  - ALRMVAL (when ALRMPTR<1:0> = `01`): Alarm Weekday and Hours Value Register
  - ALRMVAL (when ALRMPTR<1:0> = `00`): Alarm Minutes and Seconds Value Register

29.2.1 Register Mapping

To limit the register interface, the RTCC Value and RTCC Alarm Value registers are accessed through the corresponding register pointers. The RTCC Value register uses the RTCPTR bits (RCFGCAL<9:8>) to select the desired timer register pair (see Table 29-1).

The RTCPTR bits (RCFGCAL<9:8>) are automatically decremented each time the upper eight bits of the RTCVAL register are accessed by the user application. After they reach a value of '`00`', any further access to these upper eight bits by the user application has no effect on the RTCPTR bits.

Table 29-1: RTCVAL Register Mapping

RTCPTR<1:0>	RTCC Value Register Window	
	RTCVAL<15:8>	RTCVAL<7:0>
`00`	MINUTES	SECONDS
`01`	WEEKDAY	HOURS
`10`	MONTH	DAY
`11`	—	YEAR

The RTCC Alarm Value register uses the ALRMPTR bits (ALCFGRPT<9:8>) to select the desired Alarm register pair (see Table 29-2).

The ALRMPTR bits (ALCFGRPT<9:8>) are automatically decremented each time the upper eight bits of the ALRMVAL register are accessed by the user application. After they reach a value of '`00`', any further access to these upper eight bits by the user application has no effect on the ALRMPTR bits.

Table 29-2: ALRMVAL Register Mapping

ALRMPTR<1:0>	Alarm Value Register Window	
	ALRMVAL<15:8>	ALRMVAL<7:0>
`00`	MINUTES	SECONDS
`01`	WEEKDAY	HOURS
`10`	MONTH	DAY
`11`	—	YEAR

Because the 16-bit core does not distinguish between 8-bit and 16-bit read operations, reading either the upper or lower bytes decrements the ALRMPTR bits value. The same applies to the RTCPTR bits value. While writing, the RTCPTR and ALRMPTR register values decrement only when writing to the RTCVALH and ALRMVALH bytes, respectively. Writing to the RTCVALL and ALRMVALL bytes does not affect the corresponding pointer bits.

Note:	Displaying the RTCVAL or ALRMVAL register in the MPLAB® IDE Watch window causes these registers to decrement.

29.2.2 RTCC Control Registers

Register 29-1: RCFGCAL: RTCC Calibration and Configuration Register

R/W-0	U-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0
RTCEN(1)	—	RTCWREN	RTCSYNC	HALFSEC(2)	RTCOE	RTCPTR<1:0>	
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CAL<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **RTCEN:** RTCC Enable bit(1)
1 = RTCC module is enabled
0 = RTCC module is disabled

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **RTCWREN:** RTCC Value Registers Write Enable bit
1 = RTCVAL register can be written to by the user application
0 = RTCVAL register is locked out from being written to by the user application

bit 12 **RTCSYNC:** RTCC Value Registers Read Synchronization bit
1 = A rollover is about to occur in 32 clock edges (approximately 1 ms)
0 = A rollover will not occur

bit 11 **HALFSEC:** Half-Second Status bit(2)
1 = Second half period of a second
0 = First half period of a second

bit 10 **RTCOE:** RTCC Output Enable bit
1 = RTCC output is enabled
0 = RTCC output is disabled

bit 9-8 **RTCPTR<1:0>:** RTCC Value Register Pointer bits
Points to the corresponding RTCC Value register when reading the RTCVAL register. The RTCPTR bits value decrements on every access of the RTCVAL register until it reaches ‘00’. See **29.2.1 “Register Mapping”** for bit description details.

bit 7-0 **CAL<7:0>:** RTCC Drift Calibration bits
01111111 = Maximum positive adjustment; adds 508 RTCC clock pulses every one minute
•
•
•
00000001 = Minimum positive adjustment; adds 4 RTCC clock pulses every one minute
00000000 = No adjustment
11111111 = Minimum negative adjustment; subtracts 4 RTCC clock pulses every one minute
•
•
•
10000000 = Maximum negative adjustment; subtracts 512 RTCC clock pulses every one minute

Note 1: A write to this bit is only allowed when RTCWREN = 1.
2: This is a read-only bit. It is cleared to ‘0’ on a write to the lower half of the MINSEC register.

Note: The RCFGCAL register is only affected by a POR.

Register 29-2: PADCFG1: Pad Configuration Control Register

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	RTSECSEL[(1)]	PMPTTL
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-2 **Unimplemented:** Read as ‘0’

bit 1 **RTSECSEL:** RTCC Seconds Clock Output Select bit[(1)]
1 = RTCC seconds clock is selected for the RTCC pin
0 = RTCC alarm pulse is selected for the RTCC pin

bit 0 Not used by the RTCC module. See the specific device data sheet for a description of this bit.

Note 1: To enable the actual RTCC output, the RTCOE bit (RCFGCAL<10>) needs to be set.

Register 29-3: ALCFGRPT: Alarm Configuration Register

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ALRMEN	CHIME	AMASK<3:0>				ALRMPTR<1:0>	
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ARPT<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **ALRMEN:** Alarm Enable bit

1 = Alarm is enabled (cleared automatically after an alarm event whenever ARPT<7:0> = 0x00 and CHIME = 0)
0 = Alarm is disabled

bit 14 **CHIME:** Chime Enable bit

1 = Chime is enabled; ARPT<7:0> bits are allowed to roll over from 0x00 to 0xFF
0 = Chime is disabled; ARPT<7:0> bits stop once they reach 0x00

bit 13-10 **AMASK<3:0>:** Alarm Mask Configuration bits

0000 = Every half second
0001 = Every second
0010 = Every 10 seconds
0011 = Every minute
0100 = Every 10 minutes
0101 = Every hour
0110 = Once a day
0111 = Once a week
1000 = Once a month
1001 = Once a year (when configured for February 29, once every 4 years)
101x = Reserved
11xx = Reserved

bit 9-8 **ALRMPTR<1:0>:** Alarm Value Register Window Pointer bits

Points to the corresponding Alarm Value registers when reading the ALRMVALH and ALRMVALL registers. The ALRMPTR<1:0> value decrements on every read or write of ALRMVALH until it reaches ‘00’. See **29.2.1 “Register Mapping”** for bit description details.

bit 7-0 **ARPT<7:0>:** Alarm Repeat Counter Value bits

11111111 = Alarm will repeat 255 more times
•
•
•
00000000 = Alarm will not repeat

The counter decrements on any alarm event. The counter is prevented from rolling over from 0x00 to 0xFF unless CHIME = 1.

29.2.3 RTCC Value Registers

Register 29-4: RTCVAL (when RTCPTR<1:0> = 11): Year Value Register

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
YRTEN<3:0>				YRONE<3:0>			
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-4 **YRTEN<3:0>:** Binary-Coded Decimal Value of Year’s Tens Digit; contains a value from 0 to 9

bit 3-0 **YRONE<3:0>:** Binary-Coded Decimal Value of Year’s Ones Digit; contains a value from 0 to 9

Note:	A write to this register is only allowed when RTCWREN = 1.

Register 29-5: RTCVAL (when RTCPTR<1:0> = 10): Month and Day Value Register

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	MTHTEN0	MTHONE<3:0>			
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	DAYTEN<1:0>		DAYONE<3:0>			
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as ‘0’

bit 12 **MTHTEN0:** Binary-Coded Decimal Value of Month’s Tens Digit; contains a value from 0 or 1

bit 11-8 **MTHONE<3:0>:** Binary-Coded Decimal Value of Month’s Ones Digit; contains a value from 0 to 9

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-4 **DAYTEN<1:0>:** Binary-Coded Decimal Value of Day’s Tens Digit; contains a value from 0 to 3

bit 3-0 **DAYONE<3:0>:** Binary-Coded Decimal Value of Day’s Ones Digit; contains a value from 0 to 9

Note:	A write to this register is only allowed when RTCWREN = 1.

Register 29-6: RTCVAL (when RTCPTR<1:0> = `01`): Weekday and Hours Value Register

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	—	—	—	—	WDAY<2:0>		
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	HRTEN<1:0>		HRONE<3:0>			
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-11	**Unimplemented:** Read as ‘`0`’
bit 10-8	**WDAY<2:0>:** Binary-Coded Decimal Value of Weekday Digit; contains a value from 0 to 6
bit 7-6	**Unimplemented:** Read as ‘`0`’
bit 5-4	**HRTEN<1:0>:** Binary-Coded Decimal Value of Hour’s Tens Digit; contains a value from 0 to 2
bit 3-0	**HRONE<3:0>:** Binary-Coded Decimal Value of Hour’s Ones Digit; contains a value from 0 to 9

Note: A write to this register is only allowed when RTCWREN = `1`.

Register 29-7: RTCVAL (when RTCPTR<1:0> = `00`): Minutes and Seconds Value Register

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	MINTEN<2:0>			MINONE<3:0>			
bit 15							bit 8

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	SECTEN<2:0>			SECONE<3:0>			
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15	**Unimplemented:** Read as ‘`0`’
bit 14-12	**MINTEN<2:0>:** Binary-Coded Decimal Value of Minute’s Tens Digit; contains a value from 0 to 5
bit 11-8	**MINONE<3:0>:** Binary-Coded Decimal Value of Minute’s Ones Digit; contains a value from 0 to 9
bit 7	**Unimplemented:** Read as ‘`0`’
bit 6-4	**SECTEN<2:0>:** Binary-Coded Decimal Value of Second’s Tens Digit; contains a value from 0 to 5
bit 3-0	**SECONE<3:0>:** Binary-Coded Decimal Value of Second’s Ones Digit; contains a value from 0 to 9

Note: A write to this register is only allowed when RTCWREN = `1`.

29.2.4 Alarm Value Registers

Register 29-8: ALRMVAL (when ALRMPTR<1:0> = 10): Alarm Month and Day Value Register

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	MTHTEN0	MTHONE<3:0>			
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	DAYTEN<1:0>		DAYONE<3:0>			
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as '0'

bit 12 **MTHTEN0:** Binary-Coded Decimal Value of Month's Tens Digit; contains a value from 0 or 1

bit 11-8 **MTHONE<3:0>:** Binary-Coded Decimal Value of Month's Ones Digit; contains a value from 0 to 9

bit 7-6 **Unimplemented:** Read as '0'

bit 5-4 **DAYTEN<1:0>:** Binary-Coded Decimal Value of Day's Tens Digit; contains a value from 0 to 3

bit 3-0 **DAYONE<3:0>:** Binary-Coded Decimal Value of Day's Ones Digit; contains a value from 0 to 9

Note: A write to this register is only allowed when RTCWREN = 1.

Register 29-9: ALRMVAL (when ALRMPTR<1:0> = 01): Alarm Weekday and Hours Value Register

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	—	—	—	—	WDAY<2:0>		
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	HRTEN<1:0>		HRONE<3:0>			
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-11 **Unimplemented:** Read as '0'

bit 10-8 **WDAY<2:0>:** Binary-Coded Decimal Value of Weekday Digit; contains a value from 0 to 6

bit 7-6 **Unimplemented:** Read as '0'

bit 5-4 **HRTEN<1:0>:** Binary-Coded Decimal Value of Hour's Tens Digit; contains a value from 0 to 2

bit 3-0 **HRONE<3:0>:** Binary-Coded Decimal Value of Hour's Ones Digit; contains a value from 0 to 9

Note: A write to this register is only allowed when RTCWREN = 1.

Register 29-10: ALRMVAL (when ALRMPTR<1:0> = 00): Alarm Minutes and Seconds Value Register

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	MINTEN<2:0>			MINONE<3:0>			
bit 15							bit 8

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	SECTEN<2:0>			SECONE<3:0>			
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

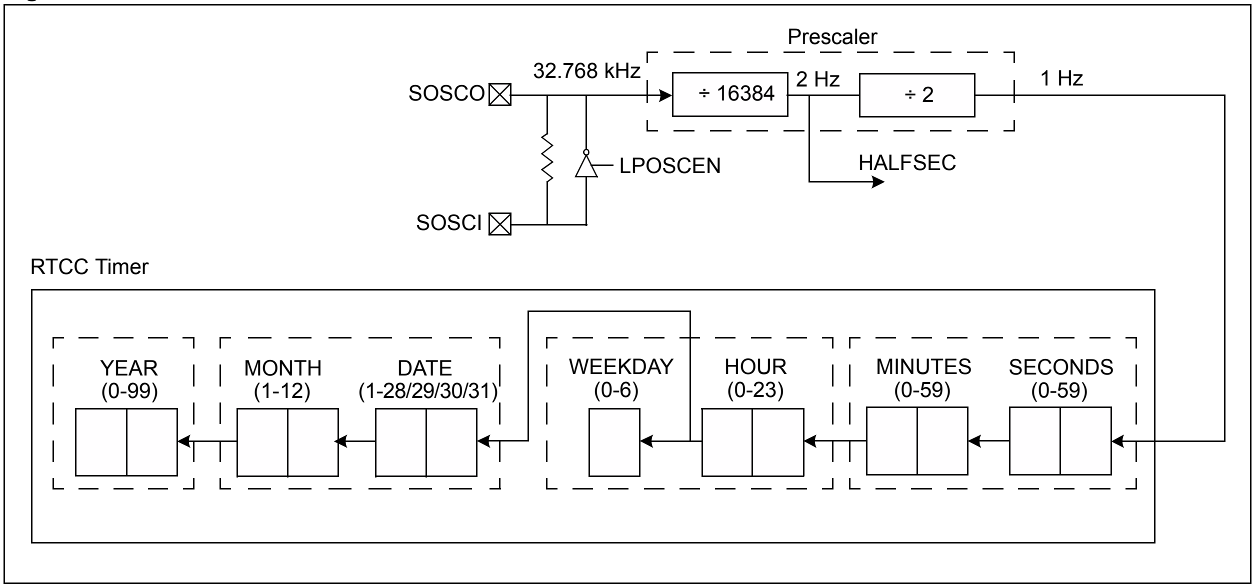
bit 15	**Unimplemented:** Read as ‘0’
bit 14-12	**MINTEN<2:0>:** Binary-Coded Decimal Value of Minute’s Tens Digit; contains a value from 0 to 5
bit 11-8	**MINONE<3:0>:** Binary-Coded Decimal Value of Minute’s Ones Digit; contains a value from 0 to 9
bit 7	**Unimplemented:** Read as ‘0’
bit 6-4	**SECTEN<2:0>:** Binary-Coded Decimal Value of Second’s Tens Digit; contains a value from 0 to 5
bit 3-0	**SECONE<3:0>:** Binary-Coded Decimal Value of Second’s Ones Digit; contains a value from 0 to 9

29.3 RTCC OPERATION

The RTCC module is intended to be clocked by an external real-time clock crystal oscillating at 32.768 kHz. The prescaler divides the crystal oscillator frequency to produce the 1 Hz update frequency for the clock and calendar. The current date and time is tracked using a 7-byte counter register that is updated once per second.

Each counter counts in BCD because it allows easy conversion to decimal digits for displaying or printing. The count sequence of the individual byte counters is illustrated in Figure 29-2. Note that the day of month and month counters roll over to one. All other counters roll over to zero. The number of days in a month and corrections for leap year are automatically adjusted. It is valid up to 2100 years. Upon initial application of power, the counters contain random information.

Figure 29-2: RTCC Timer



Note:	The Secondary Oscillator Enable bit (LPOSCEN) in the Oscillator Control register (OSCCON<1>) is set to allow the RTCC module to be clocked by the SOSC. For more details, refer to **Section 7. "Oscillator"** (DS70580) in the *"dsPIC33E/PIC24E Family Reference Manual".*

29.3.1 Writing to the RTCC Timer

The user application can configure the time and calendar by writing the desired seconds, minutes, hours, weekday, date, month and year to the RTCC registers. Under normal operation, writes to the RTCC Value registers are not allowed. Attempted writes will appear to execute normally, but the contents of the registers will remain unchanged. To write to the RTCC Value register, the RTCWREN bit (RCFGCAL<13>) must be set. Setting the RTCWREN bit allows writes to the RTCC registers. Conversely, clearing the RTCWREN bit prevents writes.

The RTCWREN bit can be cleared at any time. To set the RTCWREN bit, follow these steps:

1. Write 0x55 to NVMKEY.
2. Write 0xAA to NVMKEY.
3. Set the RTCWREN bit using a single cycle instruction.

The RTCC module is enabled by setting the RTCEN bit (RCFGCAL<15>). To set or clear the RTCEN bit, the RTCWREN bit (RCFGCAL<13>) must be set.

If the entire clock (hours, minutes and seconds) needs to be corrected, it is recommended that the RTCC module be disabled to avoid coincidental write operation when the timer increments. This prevents the clock from counting during a write to the RTCC Value register.

29
Real-Time Clock and Calendar (RTCC)

If just a single location requires modification, such as hours, minutes or seconds, the clock should not be stopped. (For example, a single field is modified to correct for daylight savings time.) Stopping the clock to modify a single location introduces an error in the timekeeping. In addition, a write to the minutes or seconds register resets the prescaler.

To write to the clock on-the-fly, the user application should wait until the RTCSYNC bit (RCFGCAL<12>) is cleared before writing to the timekeeping registers. The RTCSYNC bit is set to 32 clock edges (~1 ms) before a rollover is about to occur in the prescaler. Example 29-1 provides a sample code sequence to write to a RTCC timekeeping register.

Example 29-1: RTCC Timekeeping Register Write Operation

```
;**********************************************
; Enable RTCC Timer Access
;**********************************************
MOV     #0x55,W0
MOV     W0, NVMKEY
MOV     #0xAA,W0
MOV     W0, NVMKEY
BSET    RCFGCAL,#RTCWREN   ; Set RTCWREN bit

;**********************************************
; Disable the RTCC module
;**********************************************
BCLR    RCFGCAL,#RTCEN;    ; Clear RTCEN bit

;**********************************************
; Write to RTCC Timer
;**********************************************
MOV     RCFGCAL,w0
OR      #0x300,w0
MOV     w0,RCFGCAL         ; Set RTCPTR to 3

MOV     #0x0007,w0         ; Set Year (#0x00YY)
MOV     #0x1028,w1         ; Set Month and Day (#0xMMDD)
MOV     #0x0110,w2         ; Set Weekday and Hour (#0x0WHH)
MOV     #0x0000,w3         ; Set Minute and Second (#0xMMSS)

MOV     w0,RTCVAL
MOV     w1,RTCVAL
MOV     w2,RTCVAL
MOV     w3,RTCVAL

;**********************************************
; Enable the RTCC module
;**********************************************
BSET    RCFGCAL,#RTCEN     ; Set RTCEN bit

;**********************************************
; Disable RTCC Timer Access
;**********************************************
BCLR    RCFGCAL,#RTCWREN   ; Clear RTCWREN bit
```

Note:	The alarm must be disabled (ALRMEN = 0) while writing to real-time clock registers, otherwise a false alarm could be generated.

29.3.2 Reading the RTCC Timer

The time and calendar information is obtained by reading the appropriate register bytes. The RTCC timer is read on-the-fly. To read the current time, the RTCC timer should not be stopped. Doing so would introduce an error in its timekeeping.

As clocking of the counter occurs asynchronously to the reading of the counter, it is possible to read the counter while it is being incremented (rollover). This may result in an incorrect time reading. Therefore, to ensure a correct reading of the entire contents of the clock (or that part of interest), it must be read without a clock rollover occurring.

To read the clock on-the-fly, the user application should wait until the RTCSYNC bit (RCFGCAL<12>) becomes '0', and then read the timekeeping registers. The RTCSYNC bit is set to 32 clock edges (~1 ms) before a rollover is about to occur in the prescaler.

If the user application cannot wait for the RTCSYNC bit to become '0', the alternate method is to read the timekeeping registers twice. If the data is same both the times, it is considered to be valid. In that case, a minimum of two and a maximum of three accesses are required.

Example 29-2 provides a sample code sequence to read a RTCC timekeeping register.

Example 29-2: RTCC Timekeeping Register Read Operation

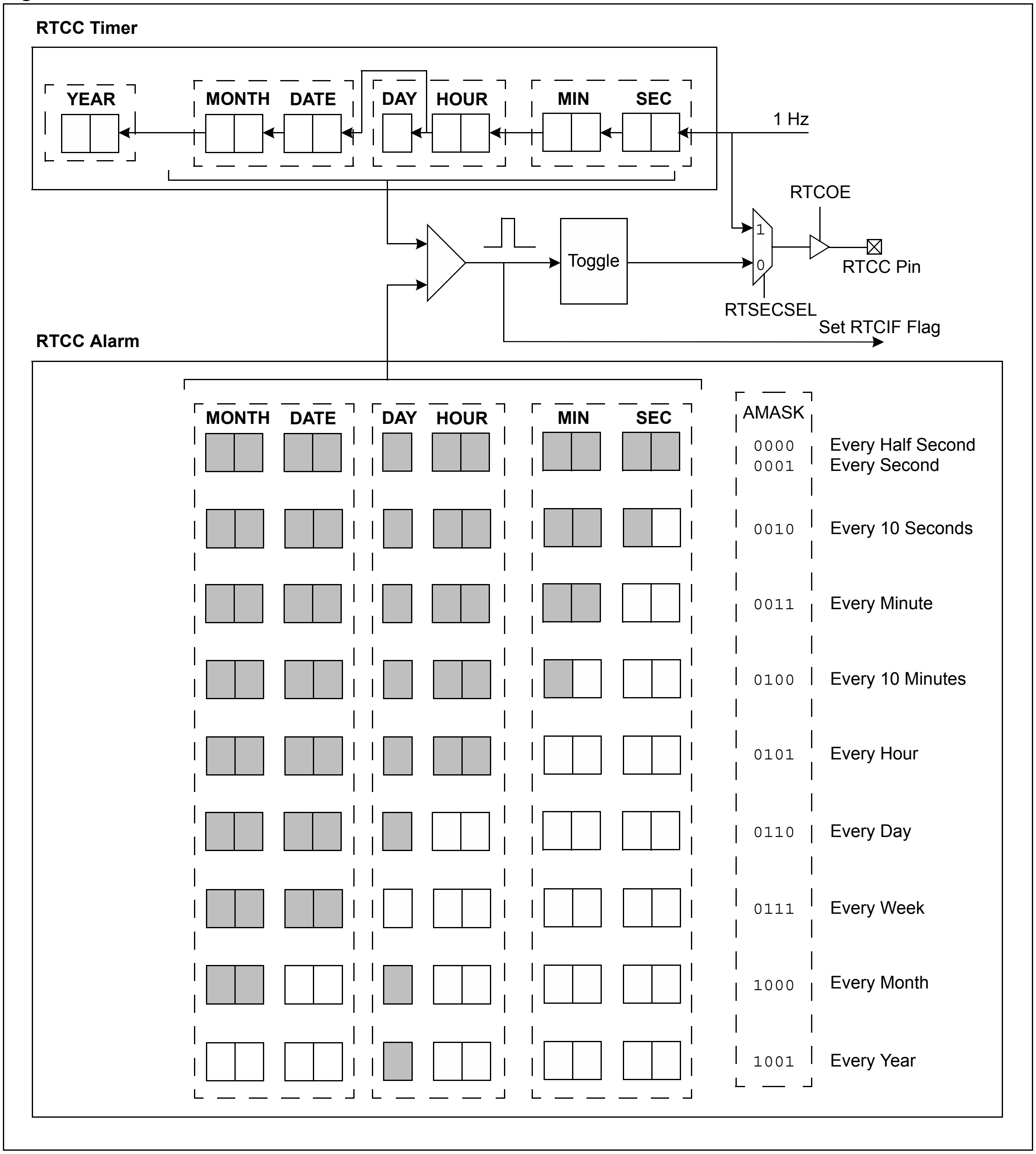
```
;**********************************************
; Wait for the RTCSYNC bit to become '0'
;**********************************************
while(RCFGCALbits.RTCSYNC == 1);

;**********************************************
; Read the RTCC timekeeping register
;**********************************************
RCFGCALbits.RTCPTR = 3;

year = RTCVAL;
month_date = RTCVAL;
wday_hour = RTCVAL;
min_sec = RTCVAL;
```

29.4 RTCC ALARM

Alarms are available to interrupt the CPU at a particular time or at periodic time intervals, such as once per minute or once per day. During each clock update, the RTCC compares the selected alarm registers with the corresponding clock registers. When a match occurs, an alarm event is generated.

The Alarm Mask bits (AMASK<3:0>) in the Alarm Configuration register (ALCFGRPT<13:10>) are used to mask registers that do not have to be compared. Figure 29-3 illustrates the alarm BCD register nibbles that are compared with the timekeeping register for different AMASK settings.

Figure 29-3: RTCC Alarm



For example, if you want to set an alarm for each morning at six o'clock (06:00:00 in 24-hour format), perform the following procedure:

1. In the Alarm Value register (ALRMVAL), load the hours byte with 06 (BCD), the minutes byte with 00 (BCD), and the seconds byte with 00 (BCD).
2. Mask off the month, date and weekday bytes and just compare hour, minute and second by setting the addressing mask bits (AMASK) to '0b0110. Each day when the time rolls over from 05:59:59, an alarm event is generated.

Note:	The alarm must be disabled (ALRMEN = 0) in the Alarm Configuration register (ALCFGRPT) while writing to ALRMVAL, otherwise it may result in a false alarm. ALRMVAL can be read at any time.

Example 29-3 provides a sample code sequence to configure the Alarm registers to generate an alarm every morning at six o'clock.

Example 29-3: Writing to Alarm Register

```
;*****************************************************
; Disable the Alarm
;*****************************************************
ALCFGRPTbits.ALRMEN = 0;

;*****************************************************
; Write to the Alarm Register (Time: 06:00:00)
;*****************************************************
ALCFGRPTbits.ALRMPTR = 2;

ALRMVAL = 0x0000;
ALRMVAL = 0x0006;
ALRMVAL = 0x0000;

;*****************************************************
; Select Alarm Mask to compare hour, minute and second
;*****************************************************
ALCFGRPTbits.AMASK = 0b0110;

;*****************************************************
; Enable the Alarm
;*****************************************************
ALCFGRPTbits.CHIME = 1;
ALCFGRPTbits.ALRMEN = 1;
```

29.4.1 Alarm Mode Selection

The alarm is enabled using the ALRMEN bit (ALCFGRPT<14>), which can generate a one-time alarm and a recurring alarm.

29.4.1.1 ONE-TIME ALARM

This alarm event is generated when the clock matches the selected alarm nibbles. The alarm is automatically disabled on an alarm event. Complete the following steps to configure a one-time alarm:

1. Clear the Chime Enable bit (CHIME) in the Alarm Configuration register (ALCFGRPT<14>).
2. Set the Alarm Repeat Counter Value bits (ARPT<7:0>) to '0'.

29.4.1.2 CHIME DISABLE

When CHIME = 0, the Alarm Repeat Counter Value bits (ARPT<7:0>) are decremented on every alarm event, and the alarm is disabled when the repeat counter becomes zero.

29.4.1.3 CHIME ENABLE

Indefinite repetition of the alarm can occur if the Chime Enable bit (CHIME) is set. When CHIME = 1, the Alarm Repeat Counter Value bits (APRT<7:0>) in the ALCFGRPT register are decremented indefinitely each time the alarm is issued.

29.4.2 Alarm Interrupt and Output

The alarm event can generate an RTCC interrupt or toggle the RTCC output pin. An RTCC interrupt is enabled using the respective RTCC Interrupt Enable bit (RTCIE). The Interrupt Priority Level bits (RTCIP<2:0>) must be written with a non-zero value for the RTCC to be a source of interrupts. For more details, refer to **Section 6. “Interrupts”** (DS70600) in the *“dsPIC33E/PIC24E Family Reference Manual”.*

In addition, an alarm event can toggle the RTCC pin, thereby generating a periodic clock at half the alarm rate. The RTCC pin is also capable of outputting the seconds clock. The user application can select between the alarm output or the seconds clock output using the RTSECSEL bit (PADCFG1<1>). When RTSECSEL = `0`, the alarm toggles the pin on every alarm event. When RTSECSEL = `1`, the seconds clock is selected.

29.5 RTCC CALIBRATION

Calibration is provided to compensate for the nominal crystal frequency and for variations in the external crystal frequency over a temperature range. Calibration is accomplished by adding or subtracting counts every one minute. Adding counts speeds up the clock and subtracting counts slows the clock. The number of pulses blanked (subtracted for negative calibration) or inserted (added for positive calibration) is set by the 8-bit signed value loaded into the RTCC Drift Calibration bits (CAL<7:0>) in the RTCC Configuration and Calibration Control register (RCFGCAL<7:0>).

Each calibration step either adds four or subtracts four oscillator cycles for every one minute (60 x 32.768 kHz = 1,966,080 oscillator cycles). This equates to ±2.034 parts per million (ppm) of adjustment per calibration step or ±5.35 seconds per month, allowing calibration of the timekeeping accuracy within three seconds per month.

Table 29-3 provides the details of the amount of adjustment for each value in the calibration register.

Table 29-3: Calibration Adjustment Values

Calibration Value (CAL<7:0>)	Adjustment Accuracy (ppm)	Time (sec/month)	Calibration Value (CAL<7:0>)	Adjustment Accuracy (ppm)	Time (sec/month)
0	0	—	-1	-2.03	-5.27
1	2.03	5.27	-2	-4.07	-10.55
2	4.07	10.55	-3	-6.1	-15.82
3	6.1	15.82	-4	-8.14	-21.09
• • •	• • •	• • •	• • •	• • •	• • •
125	254.31	659.18	-126	-256.35	-664.45
126	256.35	664.45	-127	-258.38	-669.73
127	258.38	669.73	-128	-260.42	-675

To establish the degree of calibration needed in a given application, the following method, which is specially suited to manufacturing environments, can be used:

1. Output the seconds clock (1 Hz) on the RTCC pin, measure the crystal oscillator frequency, and calculate the crystal frequency deviation in Hz.
2. Frequency Error = 32.768 kHz – Measured Frequency.
3. Calculate counter error per minute due to frequency deviation.

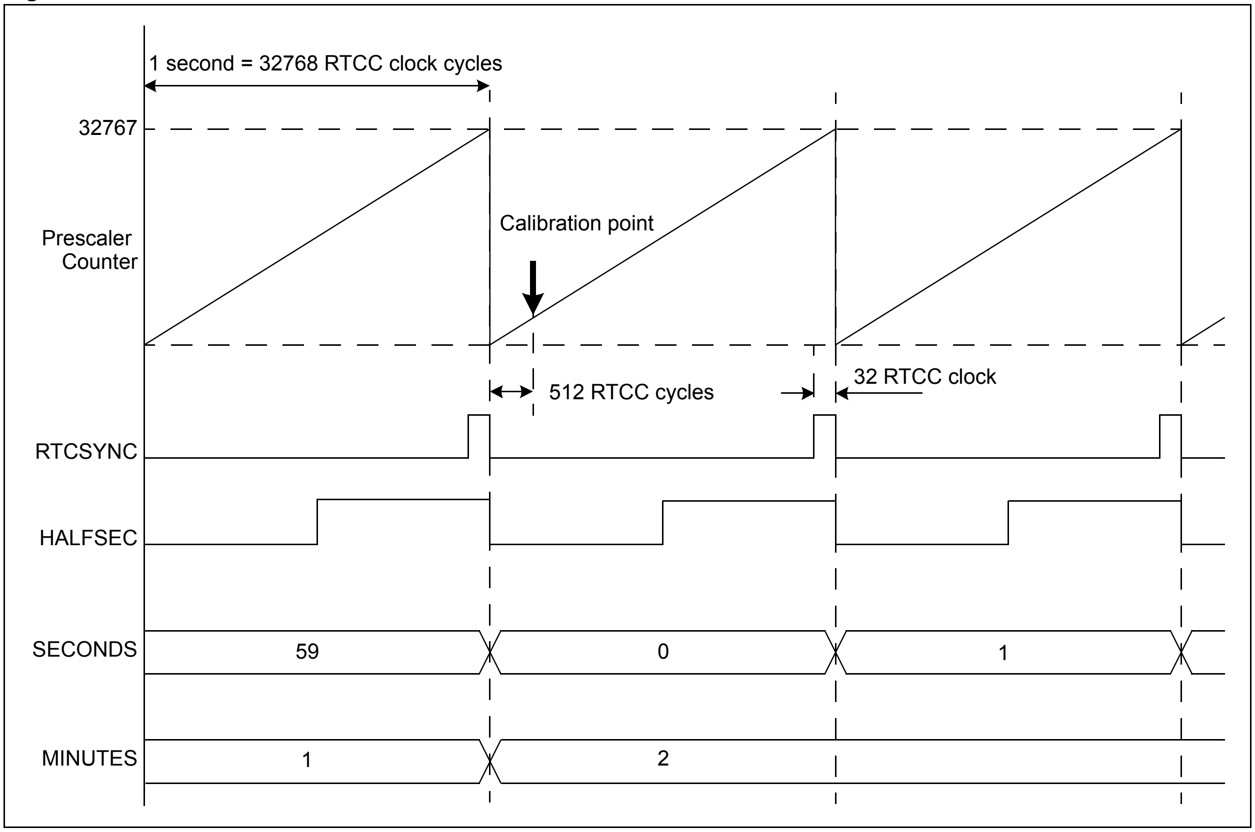
   Count Error = (Frequency Error) x 60.
4. Set the CAL<7:0> bits to (Count Error/4).

> **Note:** The user application can use a temperature sensor to adjust the calibration value to compensate for the temperature drift.

29.5.1 Writing the Calibration Value

The RTCC module performs calibration or adjusts the prescaler for every minute (see Figure 29-4). The calibration point is at the 512th clock edge following the seconds BCD counter rollover (59 to 00).

Figure 29-4: Calibration Point



At the calibration point, the RTCC module reads the 8-bit signed value (CAL<7:0>) and adjusts the prescaler counter accordingly. When the RTCC module reads the value, it should not be updated by the CPU; thereby ensuring that the RTCC module reads the correct calibration value. When the seconds byte is zero and the HALFSEC bit (RCFGCAL<11>) becomes '1', the calibration value must be updated.

Example 29-4 provides a sample code sequence to update the calibration value on-the-fly.

Example 29-4: Writing the Calibration Value

```
;****************************************************************
; Read the RTCC timer to check the seconds
; If the seconds byte is 0, wait for the HALFSEC bit to become 1
;****************************************************************
RCFGCALbits.RTCPTR = 0;
seconds = (RTCVAL & 0xFF);

if(seconds == 0)
   while(RCFGCALbits.HALFSEC == 0);

;***************************************************************
; Write to the calibration value
;***************************************************************
RCFGCALbits.CAL = calvalue;
```


29.6 RTCC OPERATION IN POWER-SAVING MODES

29.6.1 Sleep Mode

When the device enters Sleep mode, the system clock is disabled. Because the RTCC timer is clocked by the secondary oscillator, it will continue to run in Sleep mode.

If enabled, the alarm interrupt wakes up the device from Sleep mode and the following occurs:

- If the assigned priority for the interrupt is less than, or equal to, the current CPU priority, the device wakes up and continues code execution from the instruction following the `PWRSAV` instruction that initiated Sleep mode.
- If the assigned priority level for the interrupt source is greater than the current CPU priority, the device wakes up and the CPU exception process begins. Code execution continues from the first instruction of the timer Interrupt Service Routine (ISR).

For more details on power-saving modes, refer to **Section 9. "Watchdog Timer and Power-Saving Modes"** (DS70615) in the *"dsPIC33E/PIC24E Family Reference Manual"*.

29.6.2 Idle Mode

Idle mode does not affect the operation of the RTCC module.

29.7 REGISTER MAP

A summary of the registers associated with the dsPIC33E/PIC24E RTCC module is provided in Table 29-4.

Table 29-4: RTCC Register Map

File Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
ALRMVAL	Alarm Value Register Window based on ALRMPTR<1:0>																xxxx
ALCFGRPT	ALRMEN	CHIME	AMASK<3:0>				ALRMPTR<1:0>		ARPT<7:0>								0000
RTCVAL	RTCC Value Register Window based on RTCPTR<1:0>																xxxx
RCFGCAL	RTCEN	—	RTCWREN	RTCSYNC	HALFSEC	RTCOE	RTCPTR<1:0>		CAL<7:0>								0000
PADCFG1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RTSECSEL	PMPTTL	0000

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

29.8 RELATED APPLICATION NOTES

This section lists application notes that are related to this section of the manual. These application notes may not be written specifically for the dsPIC33E/PIC24E product family, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to the RTCC module are:

Title	Application Note #
No related application notes at this time.	

Note:	Please visit the Microchip web site (www.microchip.com) for additional application notes and code examples for the dsPIC33E/PIC24E family of devices.

29.9 REVISION HISTORY

Revision A (September 2009)

This is the initial released version of this document.

Revision B (June 2010)

This revision includes the following changes:

- Deleted the watermark “Untested Code – For Information Purposes Only” from the code examples
- Minor changes to the text and formatting have been incorporated throughout the document

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC^{32} logo, rfPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Octopus, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rfLAB, Select Mode, Total Endurance, TSHARC, UniWinDriver, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

Printed on recycled paper.

ISBN: 978-1-60932-288-5

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://support.microchip.com
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-6578-300
Fax: 886-3-6578-370

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

01/05/10